Part I

Setting the Context

Chapter 1 The Problem 3

Chapter 2 Hints Toward a Solution 87

This page intentionally left blank

1

The Problem

The issues surrounding digital identity management are a result of a combination of many factors

Today's digital identity crisis is the result of many independent factors, and their combined effects gave rise to the perfect storm that makes phishing and identity theft so lethally efficient. This chapter briefly revisits the evolution of online threats and reveals the complex connections by which apparently independent phenomena augment each other.

The section "The Advent of Profitable Digital Crime" explores the arms race between computer systems and security threats. From software piracy to phishing, from worms to defacing, we walk you through the early traumas that shaped the industry reactions to security problems. Vulnerabilities and attacks are described in a concrete fashion, without using technical terms. To fully appreciate the solutions presented in Chapter 2, "Hints Toward a Solution," it is important to have a solid, intuitive understanding of the issues that the industry is facing.

The section "Passwords: Ascent and Decline" provides a historical rationale for the use of passwords. Although passwords are sometimes still an acceptable solution on single machines and local networks, we expose the most prominent reasons why that credential type is sorely inefficient on the modern Internet.

The section "The Babel of Cryptography" provides a gentle introduction to concepts and terminology of modern cryptography, framing the notions as answers to the problems mentioned so far. As the explanation goes deeper into the capabilities of those tools, it becomes evident that cryptography is an important instrument, but not a silver bullet that can alone solve the problem of identity propagation. A quick glimpse at the number of the standards and products in use today will give you an idea of the challenges that prevent prompt and resilient interoperability.

The section "The Babel of Web User Interfaces" brings human behavior into the picture, showing how the tightest cryptographic protocol can be completely ineffective if its usage is not intuitive. The current Internet protocols, by their very nature, do not promote a user-friendly credential-gathering stage. Facts supporting this statement are presented, together with the most obvious and the more subtle effects on user confidence and proficiency when dealing with digital identity.

By the end of Chapter 1, it will be clear why the current situation is crying out for a strategic, long-term solution.

The Advent of Profitable Digital Crime

Today the Internet is part of our daily existence

You are sitting at the airport gate, waiting for your delayed flight to start boarding. The unexpected delay, however, leaves you all but stranded. In the seasonless atmosphere of the hall, you suffer the familiar heat from the laptop on, well, your lap: A Wi-Fi card and an adequate amount of battery power are enough to pro-

vide you access to an enormous number of resources. You can manage your mail, send your relatives instant messages (IMs), collaborate in real time with colleagues, check weather and timetables, check your bank accounts, buy goods online, translate a word you don't know, track packages, rent cars, trade shares, write blog posts, find out whether that duty free is really cheaper than online stores, trace routes, even access the recording of your favorite sitcom from the media center in your living room. Far from perceiving those activities as miracles, we already take them for granted. We are actually extremely disappointed when for some reason, say a lousy Wi-Fi provider, we can't gain access to those resources.

It's difficult to recall how life was before Web 2.0; nonetheless it is a useful exercise, and it will prove invaluable for putting into perspective the tools and motivations that animate today's bad guys of the online world.

The Dawn of Cracking

Many practices still in use today evolved in a less-connected era

Twelve years ago or so, Internet access was the privilege of a few. It was the time of universities and institutions, of Usenet, and very few companies. The Internet Relay Chat (IRC) channel #Italy# had 35 concurrent users on the most crowded days. It was the personal productivity era: Office, games, and computer-aided design (CAD) programs were the main reasons for having a personal computer on the desk. Most software was distributed via physical media, initially floppy disks and later CD-ROMs.

It all started with piracy

Piracy was probably the most common cybercrime at the time. Still, it was a sluggish shadow of today's phenomenon, forced to rely on expensive Bulletin Board Service (BBS), cracks passed by word of mouth, still-expensive CD burners, and full-fledged mail orders from a few hacker groups.

Although those illicit activities didn't really have to do with identity, they are of key importance because they incubated two cornerstones of digital crime evolution: cracking and organization.

Software offers opportunities for illicit gain

The main reason for cracking a program was the simplest: gaining access to a resource without having the right to do so. Breaking the license checks of a personal-productivity application in the 1990s meant disassembling and fiddling with a local copy running on your own computer, whereas today's nastiest attacks have to be performed without accessing the binary of the target process. In the former case, you are in the position of performing any modification. In the latter, you have to rely on known flaws of the program or discover a new one. A flaw that can be leveraged for compromising a program is known as an **exploit**.

A good part of the gain obtained from cracking had to do with satisfying narcissistic instincts, but the chance to pocket some change was not too far away.

Cracking bands were among the first to organize around the purpose of illicitly profiting from computers

The first forms of organized actions come from that time frame, too. Although access was still not widespread, the falling prices of the hardware and the rising interest in software gathered like-minded people in cracking bands, with true "hacking auditions" for membership admittance and a good dose of romantic rivalries. Again, this was very far from today's spamming behemoths and systematic phishing groups. With all the youth naiveties they may have had, however, those groups introduced an important idea: Software was a green field for illicit activities, and there was definitely a good chance to make an easy profit. Sellers of cracked software at a fraction of the price found an eager audience, especially because regulations (and enforcements) in that space were in their infancy. Gathering more crackers in groups noticeably improved their chances of gaining a margin. Many contributors meant a larger catalog of cracked

products and dramatically simplified the cracker's curse of those days: distribution.

In summary, crackers learned the following during that period:

- Software is a good that can be stolen.
- Circumventing software protections is possible.
- Coordinated action boosts profits.

This last item was particularly remarkable when you consider the fact that it was still a disconnected world.

The Vandalism and Bravado Era: Viruses and Worms

If piracy was the natural extension of the traditional compulsion to steal, we may think of computer virus writing as a form of vandalism.

A virus is a malicious program that can self-replicate

The idea of a computer virus is very old, but it gained real traction as potential hosts (programs) enjoyed widespread adoption and more distribution channels (BBCs, floppy disks, the first shareware). The bane of early system administrators and every dad who had fans of pirated games in the household, it elicited the creation of an entirely new software class: the antivirus applications.

Worms brought security threats to a global scale

If viruses weren't bad enough for shaking user's confidence in computer systems, with worms things went out of control. A worm does not need a host program. Rather, it leverages known exploits in network-enabled software for spreading from machine to machine. Email clients, instant messaging (IM) programs, file-sharing software, even low-level network protocol implementations can be leveraged as infection vectors.

ILOVEYOU demonstrated the power of leveraging the human factor

The worm phenomenon highlighted many of the techniques and the issues that can be found in modern security threats. The infamous worm ILOVEYOU, one of the worst global infections,

exploited social engineering to spread. It traveled in an email attachment named LOVE-LETTER-FOR-YOU.TXT.vbs, a name that was a strong motivator for launching the file and activating the worm. More refined forms of this technique contribute to phishing effectiveness.

Security problems on the Internet do not respect national borders

The whole ILOVEYOU affair hit the world with another key lesson, again a cornerstone in our quest for understanding today's cybercrime; whereas an event on the Internet can ripple through the economies of the entire globe, law enforcement is still bound to the principle of the sovereignty of nations. The alleged author of ILOVEYOU has been identified as a university student in the Philippines. However, shortly after the discovery, all charges related to his involvement with the worm were dropped, because at the time that kind of crime was not contemplated by any law of the Philippines justice system. (The loophole was promptly closed, but the new law was not retroactive.)

Even untrained people can do damage with the right tools

As the idea of leveraging exploits gained traction, a second tier of bad guys appeared on the scene: **script kiddies.** Publishing code that illustrates how to leverage an exploit of well-known programs became a habit for many gifted crackers. That code would be taken by less-gifted individuals and included in toolkits and utilities designed for "messing up." That would have meant, among other things, defacing websites, bringing servers to their knees via denial-of-service (DoS) attacks, attacking the machines of chat users, and even clumsy attempts at worm writing.

A Trojan is malicious software hidden in a legitimate program

Another common toy was the Trojan horse, or simply Trojan, a program that would be distributed hidden inside legitimate packages or disguised as some other kind of software (like a crack utility). A Trojan would install itself on the victim machine and listen for remote commands, to the delight of the attacker who would take control of the target computer (or, using the jargon in fashion at the time, "0wn it").

Conspiracy theorists may draw all sorts of illations from the computer virus/worm phenomenon; however, the reality is that no clear business model has been identified behind virus creation. The most plausible motivation is still sheer vandalism or the attempt to improve one's own reputation. The media contributed to feeding the aura of coolness around it, providing meticulous coverage of crackers who, after major accomplishments, get a dream job in the tech industry (fueling the dreams of armies of script kiddies).

In today's world there's little left of the narcissistic impulse that drove the first worm writers: nonetheless, their spreading model is still one of the most effective for gaining unauthorized access to less-protected PCs. Where yesterday the prize was the satisfaction of yet another life touched by the author's action, today it's acquiring yet another zombie PC, adding firepower that is at the service of greedy spammers. The move toward a business mindset becomes evident, and it became more and more manifest as we got closer to the present. The same drive favors worm-like distribution patterns for an economy of scale, but script kiddy tools can also be used for targeted attacks.

Viruses and worms influenced the way in which we think about computer security

The importance of viruses and worms in the evolution of security threads cannot be underestimated, because it was one of the central factors in shaping today's awareness of the dangers in using computer systems. Out there, there's somebody who can harm you and your business and won't hesitate to do it if you give him half a chance. This awareness is key to recognizing the need for some form of protection and the acceptance of the inevitable discomfort it brings. Installing an antivirus, remembering a password for accessing the computer, maintaining a personal firewall, not being able to attach executable files (EXEs) in email messages, are all nuisances that we would not accept if we thought there was no danger. It is a bit like allowing one extra hour for security-related lines at the airport after 9/11.

The Rush to Web 2.0 and Asset Virtualization

What we've seen so far aimed for the destruction of value and, in minor measure, for the improper acquisition of resources. It was a rough exploration; often the motivation for doing something was simply that you could. Today's world is far less naïve. Also thanks to those early bad experiences, security is being tightened up at every level and almost everything is more secure by default. Yet, we are registering the highest cybercrime rates in history. Many factors contribute to this situation, but one is certainly worth mentioning: The amount of value accessible from computer systems today grew to a point that gaining improper access to even a fraction of it is a highly profitable endeavor.

Gaining unauthorized access to the computer has always been one of the most attempted attacks

In the personal-productivity era, the valuable resource was the computer itself and the capabilities of the software it contained. Apart from the local networks (we cover local networks in detail in "Passwords: Ascent and Decline") initially limited to a relatively thin slice of white-collar workers, access to the computer was just a matter of knowing the BIOS password. It was not proper authentication but rather a very coarse form of authorization (again, see "Passwords: Ascent and Decline" and the definition of *blind credentials*). The same can be said for later uses of this security mechanism, such as password protecting Office documents or Zip archives. Every resource was at the complete disposal of the computer owner, with the exception of licensed software; in that case, at least in the installation phase, you had to provide some form of proof of purchase (such as the still-popular serial number, entered at installation time). There was not much to be stolen, and there was no way of doing it without sitting in front of the machine.

The advent of online services introduced a new kind of good to steal

Things started to change as computers gained access to new classes of resources that were impossible to have in local form. Among the first examples were the mail services and the connectivity provider. For the record, this is one of the first moments in which the consumer started to project his identity; gaining